A code review is an assessment of code that can be performed as a self-assessment or with peer’s code. Code reviews can identify functional errors that may not be captured by just compiling the code. By taking a step back to review the code the reviewer can provide feedback for enhancement. Code reviews can also be used to share knowledge to other developers by showing how the code functions. The purpose is to identify gaps in the structure. The review can identify opportunities for improvement that could make the code more reliable and robust. Code reviews can occur in various steps in the software development life cycle. The code to be reviewed could be a completely new feature or it could be a review that occurs after customer feedback. A best practice that I have found helpful is to use a checklist to identify specific areas for improvement. Having a checklist allows the reviewer to search for specific characteristics or features. Reviewing the code in small sections is something else I would recommend. Having a plan or making recommendations for fixes is also a good practice. The overall goal in a code review is to gain constructive feedback for improvement. If I had a peer review my code and there was little to no feedback or suggestions for improvements, this would mean that either the code is close to perfect, or the review was not very through. It would be beneficial to me to have suggestions for improvement even if I didn’t agree rather than feedback that could not be used for improvement.